

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号  
特開2000-207576  
(P2000-207576A)

(43)公開日 平成12年7月28日(2000.7.28)

(51) Int.Cl.<sup>7</sup> 識別記号 F I テーマコード(参考)  
 G 0 6 T 15/00 G 0 6 F 15/72 4 5 0 A 5 B 0 5 0  
 17/00 15/62 3 5 0 A 5 B 0 8 0  
 15/50 15/72 4 6 5

審査請求 未請求 請求項の数9 OL (全 9 頁)

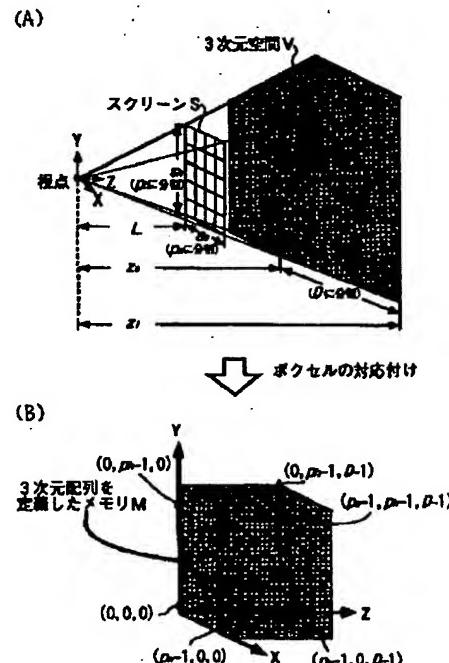
(21)出願番号	特願平11-5992	(71)出願人	000004226 日本電信電話株式会社 東京都千代田区大手町二丁目3番1号
(22)出願日	平成11年1月13日(1999.1.13)	(72)発明者	松岡 裕人 東京都新宿区西新宿三丁目19番2号 日本 電信電話株式会社内
		(72)発明者	原田 育生 東京都新宿区西新宿三丁目19番2号 日本 電信電話株式会社内
		(74)代理人	100087848 弁理士 小笠原 吉義 (外1名)

(54)【発明の名称】 画像処理方法、画像処理装置、および画像処理プログラムを記録した記録媒体

(57) 【要約】

**【課題】** レイトレーシング等のレンダリング手法を用いる画像処理において、必要とするメモリ容量が少なく、かつエリアシングを生じないボクセル分割によるレンダリングを実現する。

【解決手段】 視点を原点とし、表示対象となる3次元空間Vの3軸をX軸、Y軸、Z軸、および表示するスクリーンSがZ軸に垂直にあるとし、3次元空間Vをスクリーン平面に平行に、ある整数D個に分割し、さらに視点を通りスクリーンSのピクセルの境界を通る平面で分割して四角錐台を生成し、それらの四角錐台を3次元配列メモリMに割当て、そのメモリM上で視点からの光線と物体との交点を探索して交差判定を行う。



## 【特許請求の範囲】

【請求項1】 レイトレーシング等のレンダリング手法を用いる画像処理方法において、視点を原点とし、表示対象となる3次元空間Vの3つの軸をX軸、Y軸、Z軸、表示するスクリーンSが3次元空間VのZ軸に垂直にあるとし、3次元空間Vをスクリーン平面に平行に、ある整数D個に分割し、さらに、視点を通りスクリーンSのピクセルの境界を通る平面で分割して四角錐台を生成し、それら四角錐台を3次元配列を定義したメモリMにそれぞれ割り当て、そのメモリM上で視点からの光線と物体との交点を探索して交差判定を行うことを特徴とする画像処理方法。

【請求項2】 請求項1記載の画像処理方法において、3次元空間VのZ軸の最小値を $z_{\min}$ 、最大値を $z_{\max}$ とし、1から(D-1)までの整数をnとするとき、前記3次元空間Vを、次式の

$$z = z_{\min} + (1 - (1 - z_{\min} / z_{\max}) \times D \times n)$$

を満たす平面で分割することを特徴とする画像処理方法。

【請求項3】 請求項1または請求項2記載の画像処理方法において、スクリーンSのX軸の長さを $s_x$ 、ピクセルの数を $p_x$ 、Y軸の長さを $s_y$ 、ピクセルの数を $p_y$ 、視点との距離をLとし、表示対象となる3次元カメラ座標系で定義された一つあるいは複数の多角形Pの頂点( $x_v, y_v, z_v$ )を、次式の

$$x_n = p_x \times s_x \times L \times x_v / z_v$$

$$y_n = p_y \times s_y \times L \times y_v / z_v$$

$$z_n = (1 - z_v / z_{\max}) \div (1 - z_v / z_{\min}) \times D$$

(ただし、 $x_n, y_n, z_n$ は整数化する)に従って3次元配列を定義したメモリMの座標( $x_n, y_n, z_n$ )に変換することによって得られるメモリMの座標系における多角形Pの外周および内面に多角形Pの描画情報を書き込み、スクリーンSの任意の1ピクセルcの座標( $x_c, y_c$ )の描画データを、メモリ座標( $x_c, y_c, 0$ )を始点、方向ベクトル(0, 0, 1)を視点からの光線としてメモリ空間内を探索し、表示対象の多角形Pとの交点を求めてピクセルの色彩データを求め、さらにスクリーンS内のすべてのピクセルに対して同様に計算して3次元画像を生成することを特徴とする画像処理方法。

【請求項4】 メモリ書き込み回路、メモリ、レイトレーシング回路からなり、スクリーン平面に表示する3次元画像を生成する画像処理装置であって、前記メモリ書き込み回路は、スクリーン平面がZ軸に垂直となるように表示対象の3次元空間VのXYZ座標系をとり、3次元空間Vをスクリーン平面に平行なD個(Dはある整数)の平面で分割し、さらに視点およびスクリーンのピクセルの境界を通る平面で分割して四角錐台を生成し、それら四角錐台を3次元配列を定義した前記メモリに割り当てる描画領域算出回路を含み、前記レイトレーシン

グ回路は、前記メモリ上で視点からの光線と表示対象の物体との交点を探索して交差判定を行う交差判定回路を含むことを特徴とする画像処理装置。

【請求項5】 請求項4記載の画像処理装置において、前記描画領域算出回路は、3次元空間VのZ座標値の最小値を $z_{\min}$ 、最大値を $z_{\max}$ とし、1からD-1までの整数をnとするとき、前記3次元空間Vを、次式の  

$$z = z_{\min} + (1 - (1 - z_{\min} / z_{\max}) \div D \times n)$$

を満たすD個の平面で分割するものであることを特徴とする画像処理装置。

【請求項6】 請求項4または請求項5記載の画像処理装置において、視点を原点として前記3次元空間VのXYZ座標系をとり、スクリーンのX軸の長さを $s_x$ 、ピクセルの数を $p_x$ 、Y軸の長さを $s_y$ 、ピクセルの数を $p_y$ 、視点との距離をLとするととき、前記メモリ書き込み回路は、3次元空間V内の点( $x_v, y_v, z_v$ )を、次式の

$$x_n = p_x \times s_x \times L \times x_v / z_v$$

$$y_n = p_y \times s_y \times L \times y_v / z_v$$

$$z_n = (1 - z_v / z_{\max}) \div (1 - z_v / z_{\min}) \times D$$

(ただし、 $x_n, y_n, z_n$ は整数化する)に従って、3次元配列を定義したメモリの座標系( $x_n, y_n, z_n$ )に変換する座標変換回路と、表示対象の1つあるいは複数の多角形Pの頂点に対する座標変換結果に基づいて、メモリの座標系における多角形Pの外周および内面に多角形Pの描画データを書き込む描画領域算出回路とを含み、前記レイトレーシング回路は、スクリーンの任意の1ピクセル( $x_c, y_c$ )に対する光路をメモリ座標( $x_c, y_c, 0$ )を始点、方向ベクトル(0, 0, 1)として算出する光路演算回路と、算出された光路に従ってメモリ内を探索し、表示対象の多角形Pとの交点を算出する交差判定回路と、算出された交点とメモリ上の描画情報から前記ピクセルの色彩データを算出する色彩データ演算回路とを含むことを特徴とする画像処理装置。

【請求項7】 レイトレーシング等のレンダリング手法を用いる画像処理を計算機で行うためのプログラムを記録した記録媒体であって、視点を原点とし、表示対象となる3次元空間Vの3つの軸をX軸、Y軸、Z軸、表示するスクリーンSが3次元空間VのZ軸に垂直にあるとし、3次元空間VをスクリーンS平面に平行に整数D個に分割し、さらに、視点を通りスクリーンSのピクセルの境界を通る平面で分割して四角錐台を生成し、それら四角錐台を3次元配列を定義したメモリMにそれぞれ割り当て、そのメモリM上で視点からの光線と物体との交点を探索して交差判定を行う処理を、計算機に実行させるプログラムを記録したことを特徴とする画像処理プログラムを記録した記録媒体。

【請求項8】 請求項7記載の画像処理プログラムを記録した記録媒体において、前記画像処理プログラムは、

3次元空間VのZ軸の最小値を $z_1$ 、最大値を $z_2$ とし、1から(D-1)までの整数をnとするとき、前記3次元空間Vを、次式の

$z = z_1 + (1 - (1 - z_1 / z_2)) \times D \times n$

を満たす平面で分割する処理を、計算機に実行させるプログラムを含むことを特徴とする画像処理プログラムを記録した記録媒体。

【請求項9】 請求項7または請求項8記載の画像処理プログラムを記録した記録媒体において、前記画像処理プログラムは、スクリーンSのX軸の長さを $s_x$ 、ピクセルの数を $p_x$ 、Y軸の長さを $s_y$ 、ピクセルの数を $p_y$ 、視点との距離をLとし、表示対象となる3次元カメラ座標系で定義された一つあるいは複数の多角形Pの頂点 $(x_v, y_v, z_v)$ を、次式の

$$x_n = p_x \times s_x \times L \times x_v \div z_v$$

$$y_n = p_y \times s_y \times L \times y_v \div z_v$$

$$z_n = (1 - z_v \div z_v) \div (1 - z_1 \div z_2) \times D$$

(ただし、 $x_v, y_v, z_v$ は整数化する)に従って、3次元配列を定義したメモリMの座標 $(x_n, y_n, z_n)$ に変換することによって得られるメモリMの座標系における多角形Pの外周および内面に多角形Pの描画情報を書き込み、スクリーンSの任意の1ピクセルcの座標 $(x_c, y_c)$ の描画データを、メモリ座標 $(x_c, y_c, 0)$ を始点、方向ベクトル $(0, 0, 1)$ を視点からの光線としてメモリ空間内を探索し、表示対象の多角形Pとの交点を求めてピクセルの色彩データを求め、さらにスクリーンS内のすべてのピクセルに対して同様に計算して3次元画像を生成する処理を、計算機に実行させるプログラムを含むことを特徴とする画像処理プログラムを記録した記録媒体。

#### 【発明の詳細な説明】

##### 【0001】

【発明の属する技術分野】本発明は、3次元描画システムで用いられ、特にレイトレーシング等のレンダリング手法の交点算出手法およびその処理を用いた画像処理技術に関するものである。

##### 【0002】

【従来の技術】3次元画像を作成するレンダリング手法として、レイトレーシング法がある。図7は、レイトレーシング手法を説明するための図である。レイトレーシング法は、図7に示すように、視点からスクリーン上の画素を通過する光線を発生させ、すべての物体に対して交差する点を調べてレンダリングを行う。この手法では、物体と光線との交差判定回数が膨大となり、処理時間が大きくなるという問題がある。この処理時間を軽減する手法として、図8に示すように、3次元空間を小立方体(ボクセル)に分割し、それらを3次元配列を定義したメモリ空間に対応付けして交差判定をする手法がある。すべての物体をそのメモリに書き込み、光線に沿ってメモリ空間を探索すれば、交差判定に要する処理時間

を軽減することができる。

【0003】しかし、この従来のボクセル分割手法では、必ずしも光線がボクセルの中心を通るわけではなく、エリアシングが生じ、画質が劣化する問題がある。図9は、この画質が劣化する問題を説明するための図であって、YZ平面から見た従来のボクセル分割手法を示す図である。図9のように、視点からスクリーンSのピクセルの中心を通る光線1は、ボクセルAと交差しており、ボクセルAのデータでレンダリングを行う。しかし、ボクセルAとボクセルBの描画データは実際には連続しており、光線1との交点の実際の描画データは、ボクセルAとボクセルBの間のデータであり、その差が画像の歪みとなる。これを回避する手法として、光線2や光線3等の複数の光線も計算してレンダリングを行う手法があり、これによれば画質は改善されるが、処理時間が大きくなるという問題がある。

【0004】また、上記の手法では、視点から遠く離れば離れる程、X軸方向およびY軸方向のボクセル数が増える。このため、必要とされる3次元配列を定義するメモリ容量も大きくなるという問題もある。

##### 【0005】

【発明が解決しようとする課題】上述したように、ボクセルを用いたレイトレーシング手法は演算処理量を抑えられるが、エリアシングが生じて画質が劣化し、また、処理に必要となるメモリ容量も大きい。本発明の目的は、必要とするメモリ容量が少なく、かつエリアシングが生じないボクセル分割法によるレンダリングを実現することにある。

##### 【0006】

【課題を解決するための手段】本発明は、レイトレーシング等のレンダリング手法を用いる画像処理において、視点を原点とし、表示対象となる3次元空間Vの3つの軸をX軸、Y軸、Z軸、表示するスクリーンSが3次元空間VのZ軸に垂直にあるとし、3次元空間Vをスクリーン平面に平行にある整数D個に分割し、さらに、視点を通りスクリーンのピクセルの境界を通る平面で分割して四角錐台を生成し、それら四角錐台を3次元配列を定義したメモリMにそれぞれ割り当て、そのメモリM上で視点からの光線と物体との交点を探して交差判定を行う画像処理方法である。

【0007】本発明では、レイトレーシング等のレンダリング手法において、視点からスクリーンのピクセルの中心を通る光線が、常にボクセルの中心を通過するよう分割して表示対象の多角形と光線との交差を判定し描画するため、エリアシングが生じず、歪みの少ない画像を高速に生成することができる。また、視点から遠い場所でもX軸およびY軸方向のボクセル数はスクリーンのピクセル数に等しいため、処理に必要とするメモリ容量を低く抑えることができる。

##### 【0008】

【発明の実施の形態】〔実施の形態1〕図1に、請求項1に記載する画像処理方法の例を示す。レイトレーシング等のレンダリング手法を用いる画像処理において、視点を原点とし、表示対象となる3次元空間Vの3つの軸をX軸、Y軸、Z軸、表示するスクリーンSが3次元空間VのZ軸に対し垂直にあるとする。

【0009】図1(A)に示すように、3次元空間VをスクリーンSと平行に、ある整数D個に分割し、さらに、視点を通りスクリーンSのピクセルの境界を通る平面で分割して四角錐台を生成する。それらの四角錐台を、図1(B)に示すような3次元配列を定義したメモリMにそれぞれ割り当てる。そして、レイトレーシング等の処理で行う、視点からの光線と3次元空間V内の物体との交差判定に、そのメモリMを用いる。3次元空間V内のすべての表示対象となる物体をメモリM上に展開しているので、光線に沿ってメモリM内を探索すれば、3次元空間Vの物体との交差判定を高速に行なうことができる。

【0010】また、3次元空間Vは、視点とスクリーンSのピクセル間の境界を通る平面で分割し、ボクセルを形成している。したがって、表示対象となる3次元空間Vを本手法でボクセル化すれば、視点からスクリーンSのピクセルの中心を通った光線はボクセルの中心を通る。

【0011】図2は、YZ平面から見た本ボクセル化手法によりボクセルに分割した例を示す。図2から明らかなように、スクリーンの中心を通った光線はボクセルの中心を通ることがわかる。このように、光線がボクセルの中心を通るため、光線と交差したボクセルの情報のみで描画を行ってもエリシングは生じない。従来のボクセル分割による手法では、高品質な画像を得るために複数の光路を探索する必要があったが、本手法によれば、1\*

\* つの光路の探索で、高品質な画像を得ることができる。

【0012】〔実施の形態2〕図3に、請求項2に係る画像処理方法の例を示す。この図3では、3次元空間VのZ軸の最小値を $z_{\min}$ 、最大値を $z_{\max}$ とし、1からD-1までの整数をnとすると、

【0013】

【数1】

$$z = \frac{z_t}{1 - (1 - \frac{z_t}{z_l}) \cdot \frac{n}{D}} \quad (式1)$$

【0014】を満たす平面で分割したときのYZ平面から見た様子を示している。

【0015】図4は、Z軸方向を等分に分割した場合のボクセルをYZ平面から見た図である。この図4のように、 $z_{\min}$ から $z_{\max}$ の区間にD個に等分に分けてメモリMに割り当たる場合、3次元空間V上での反射光等の経路は、メモリM上では曲線となり、光路計算が複雑となる。これに対し、図3に示すように(式1)を満たす平面で分割した場合には、3次元空間V上の直線は、対応するメモリM上でも直線となる。

【0016】以下、数式を用いて直線となることを証明する。

【0017】スクリーンSのX軸の長さを $s_x$ 、ピクセルの数を $p_x$ 、Y軸の長さを $s_y$ 、ピクセルの数を $p_y$ 、視点との距離をlとし、3次元空間V上の1点v<sub>1</sub>(x<sub>v1</sub>, y<sub>v1</sub>, z<sub>v1</sub>)がメモリM上の1点m<sub>1</sub>(x<sub>m1</sub>, y<sub>m1</sub>, z<sub>m1</sub>)に対応するとき、v<sub>1</sub>とm<sub>1</sub>には次の関係が成り立つ。

【0018】

【数2】

$$x_m = \frac{p_x \cdot L}{s_x} x_v, y_m = \frac{p_y \cdot L}{s_y} y_v, z_m = D \cdot \frac{\frac{1 - \frac{z_t}{z_l}}{z_t}}{1 - \frac{z_t}{z_l}} \quad (式2)$$

【0019】ここで、3次元空間V内の3点v<sub>1</sub>

<sub>1</sub>(x<sub>v1</sub>, y<sub>v1</sub>, z<sub>v1</sub>), v<sub>2</sub>(x<sub>v2</sub>, y<sub>v2</sub>, z<sub>v2</sub>), v<sub>3</sub>(x<sub>v3</sub>, y<sub>v3</sub>, z<sub>v3</sub>)が直線上にあるとき、

※【0020】

【数3】

$$\frac{x_{v3} - x_{v1}}{x_{v2} - x_{v1}} = \frac{y_{v3} - y_{v1}}{y_{v2} - y_{v1}} = \frac{z_{v3} - z_{v1}}{z_{v2} - z_{v1}} \quad (式3)$$

【0021】が成り立つ。まず、yとzの項について展開すると、

★ 【数4】

$$y_{v3} \cdot z_{v2} - y_{v2} \cdot z_{v3} - y_{v2} \cdot z_{v1} + y_{v3} \cdot z_{v1} + y_{v1} \cdot z_{v3} - y_{v1} \cdot z_{v2} = 0 \quad (式4)$$

【0023】となる。(式4)のy<sub>v1</sub>, y<sub>v2</sub>, y<sub>v3</sub>の項に(式2)を代入すると、

【0024】

50 【数5】

$$\frac{s_v}{p_w \cdot l} (y_{m3}z_{v3}z_{v2} - y_{m3}z_{v3}z_{v1} - y_{m2}z_{v2}z_{v3} + y_{m2}z_{v2}z_{v1} + y_{m1}z_{v1}z_{v3} - y_{m1}z_{v1}z_{v2}) = 0$$

$$\frac{y_{m3} - y_{m3}}{z_{v1} - z_{v2}} - \frac{y_{m2}}{z_{v1}} + \frac{y_{m2}}{z_{v3}} + \frac{y_{m1}}{z_{v2}} - \frac{y_{m1}}{z_{v3}} = 0 \quad (\text{式5})$$

【0025】となる。(式5)の $z_{v1}$ ,  $z_{v2}$ ,  $z_{v3}$ の項 \* 【0026】

に(式2)を代入すると。 \* 【数6】

$$\frac{1}{z_s} (y_{m3} - y_{m3} - y_{m2} + y_{m2} + y_{m1} - y_{m1})$$

$$-\frac{1}{D \cdot z_s} \left( 1 - \frac{z_s}{z_i} \right) (y_{m3}z_{m2} - y_{m3}z_{m1} - y_{m2}z_{m3} + y_{m2}z_{m1} + y_{m1}z_{m3} - y_{m1}z_{m2}) = 0$$

$$\frac{y_{m3} - y_{m1}}{y_{m2} - y_{m1}} = \frac{z_{m3} - z_{m1}}{z_{m2} - z_{m1}} \quad (\text{式6})$$

【0027】が得られる。同様に、(式3)のxとzに \* 【0028】

関して計算すると。 \* 【数7】

$$\frac{x_{m3} - x_{m1}}{x_{m2} - x_{m1}} = \frac{y_{m3} - y_{m1}}{y_{m2} - y_{m1}} = \frac{z_{m3} - z_{m1}}{z_{m2} - z_{m1}} \quad (\text{式7})$$

【0029】が得られ、3次元空間V上の直線を(式2)を用いてメモリM上の座標に変換しても、直線が保たれることがわかる。

【0030】したがって、本変換手法を用いれば、反射光や屈折光等と物体との交差判定をメモリM上での直線を用いて行うことができ、高速に判定することができる。

【0031】〔実施の形態3〕図5は、請求項3に係るボクセル化の手法を用いたレンダリング方法を実現する処理フローチャートである。

【0032】まず、表示対象の多角形Pのジオメトリ処理(カメラ座標系に変換)を行い(ステップS1)，表示対象となる3次元カメラ座標系で定義された一つあるいは複数の多角形Pの頂点を、先に示した(式2)を用いて3次元配列を定義したメモリMの座標系に変換する(ステップS2)。メモリMの座標系で表された多角形Pの外周および内面にあたるメモリMの座標に、多角形Pの描画情報を書き込む(ステップS3)。すべての多角形のメモリへの書き込みが完了したら(ステップS4)、レイトレーシングによる描画処理を行う。

【0033】スクリーンS上の各ピクセルに対するレイトレーシング処理の初期設定として、スクリーンS上の1ピクセルcの座標を $(x_c, y_c)$ とし、メモリ座標 $(x_c, y_c, 0)$ を光線の開始点E,  $(0, 0, 1)$ を光線の方向ベクトルVとする(ステップS5)。ピクセルcの色彩情報は、視点からの光線の方向ベクトルVをもとにメモリ空間内を探索して、表示対象の多角形P

との交点を求める(ステップS6～S7)。そして、その交点の多角形Pの描画情報をもとに色彩情報を計算し、スクリーンへの書き込みを行う(ステップS8)。場合により、その交点からの反射光や屈折光等も計算してピクセルcの色彩情報を加える(ステップS9～S12)。スクリーン内のすべてのピクセルに対して、全スクリーンの描画が完了するまで同様に計算することにより(ステップS13)，3次元画像を生成することができる。

【0034】〔実施の形態4〕図6は、請求項4に係る画像処理装置のブロック構成例を示す図である。本装置は、多角形情報書き込み回路10、3次元配列を定義した3次元配列メモリ20、レイトレーシング処理回路30からなる。

【0035】多角形情報書き込み回路10は、多角形の頂点情報を、前記(式2)に従ってメモリ座標系に変換する座標変換回路11と、与えられた多角形内のメモリ座標を計算し、メモリにアドレスとして与える描画領域算出回路12からなり、多角形の描画情報を3次元配列メモリ20に書き込む。座標変換回路11は四則演算器で構成され、描画領域算出回路12はスキャンライン手法等を用いて実現することができる。

【0036】3次元配列メモリ20への書き込みが終わると、レイトレーシング処理回路30により、3次元配列メモリ20の情報から、レイトレーシング手法により3次元画像を生成し、ビデオメモリ40に書き込む。このレイトレーシング処理回路30は、光路演算回路31

により光路に沿ったメモリアドレスを生成し、光線上に多角形があるかを交差判定回路32で判定する。交差している場合、3次元配列メモリ20の描画情報をもとに色彩データ演算回路33によりビデオメモリ40に書き込む色彩データを演算する。また、その交点の反射率や屈折率のデータをもとに、反射光・屈折光演算回路34により光線ベクトルを求め、光路演算回路31に与え、同様に反射光や屈折光の色彩データを求める。なお、これらの演算回路は論理回路を組み合わせた集積回路等で実現することができる。

【0037】また、汎用のCPUやDSP(Digital Signal Processor)等の計算機でも、これらの処理を行うことができる。これらの処理を行うためのプログラムは、計算機が読み取り可能な可搬媒体メモリ、半導体メモリ、ハードディスクなどの適当な記録媒体に格納することができる。

#### 【0038】

【発明の効果】以上述べたように、本発明を用いることにより、レイトレンダリング等のレンダリング手法において、視点からスクリーンのピクセルの中心を通る光線が、常にボクセルの中心を通過するように分割して、表示対象の多角形と光線との交差を判定し描画するため、エリアシングが生じず、歪みの少ない画像を高速に生成することができる。また、視点から遠い場所でもX軸およびY軸方向のボクセル数はスクリーンのピクセル数に等しいため、処理に必要とするメモリ容量を低く抑えることができる。

#### 【図面の簡単な説明】

\* 【図1】画像処理方法の第1の実施の形態を説明するための図である。

【図2】YZ平面から見た第1の実施の形態のボクセル化手法を示す図である。

【図3】画像処理方法の第2の実施の形態を説明するための図である。

【図4】Z軸方向を等分に分割した場合のボクセルをYZ平面から見た図である。

【図5】第3の実施の形態のボクセル化手法を用いたレンダリング方法を実現する処理フローチャートである。

【図6】画像処理装置のブロック構成例を示す図である。

【図7】レイトレンダリング手法説明図である。

【図8】従来のボクセル分割手法を示す図である。

【図9】YZ平面から見た従来のボクセル分割手法を示す図である。

#### 【符号の説明】

S スクリーン

V 3次元空間

20 M 3次元配列を定義したメモリ

L 視点とスクリーンとの距離

D ある整数

s<sub>x</sub> スクリーンのX軸の長さ

s<sub>y</sub> スクリーンのY軸の長さ

p<sub>x</sub> スクリーンのX軸方向のピクセルの数

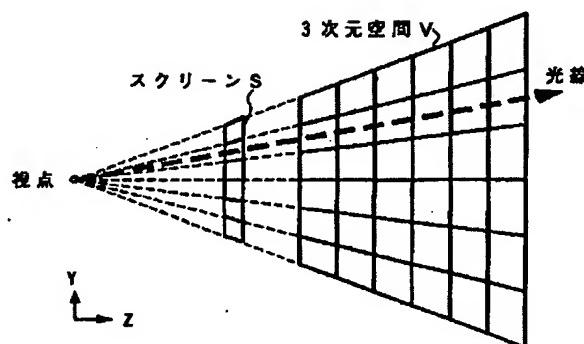
p<sub>y</sub> スクリーンのY軸方向のピクセルの数

z<sub>min</sub> Z軸の最小値

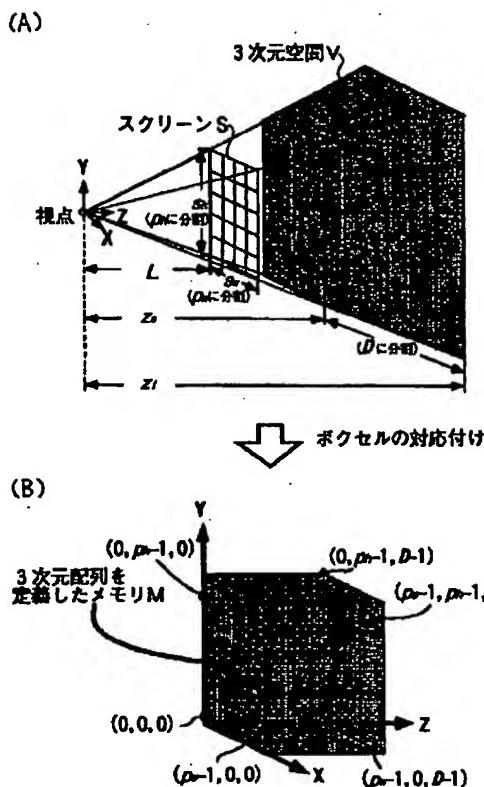
z<sub>max</sub> Z軸の最大値

\*

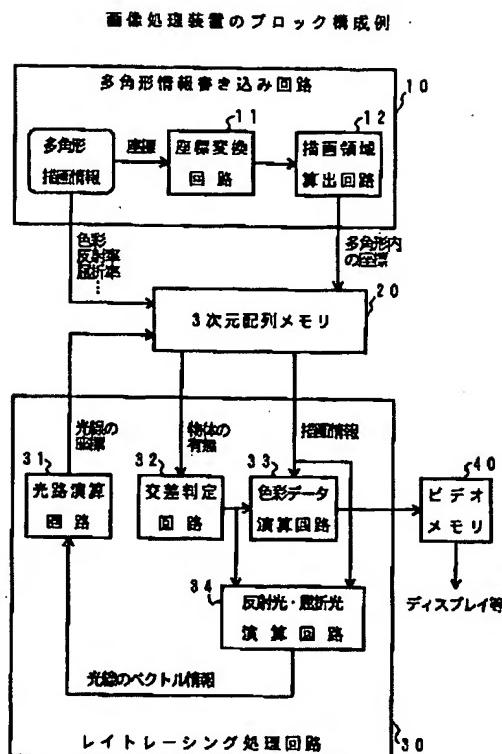
【図2】



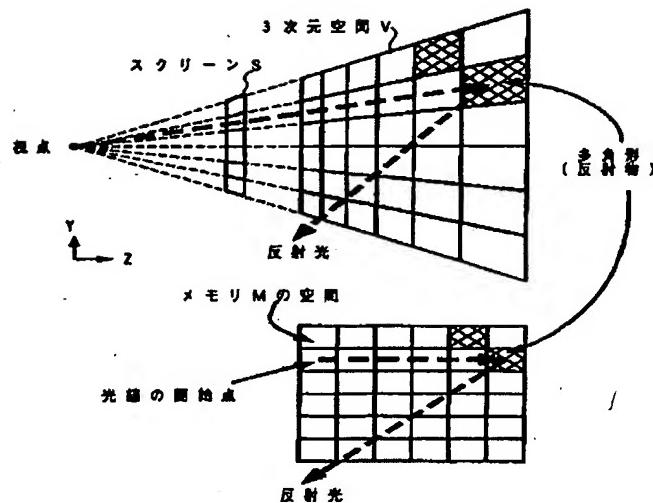
【図1】



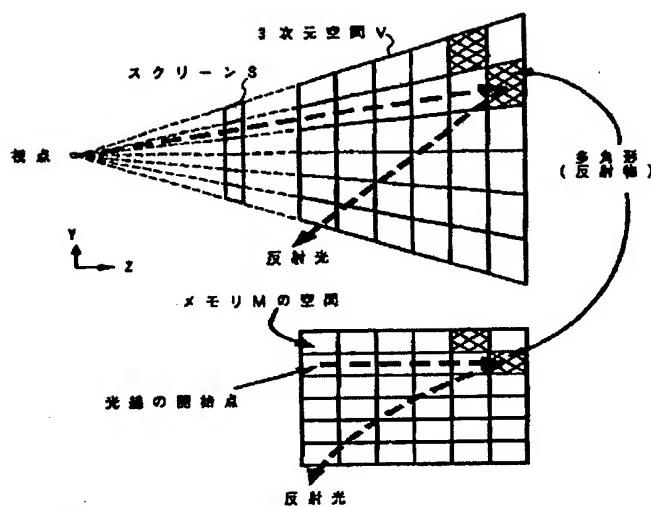
【図6】



【図3】

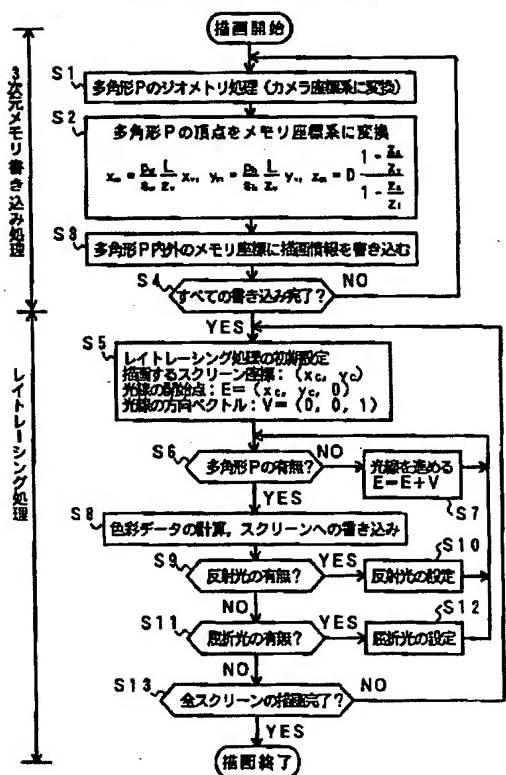


【図4】



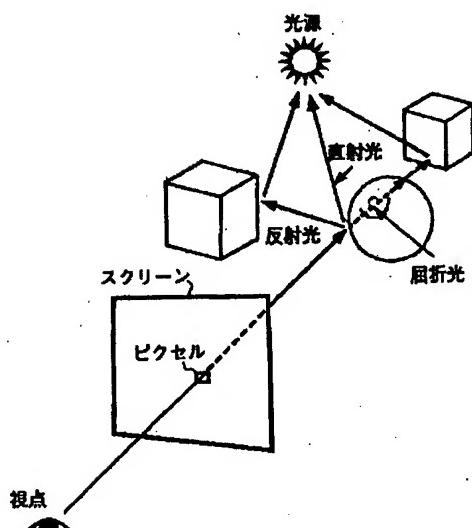
【図5】

レンダリング方法の処理フローチャート



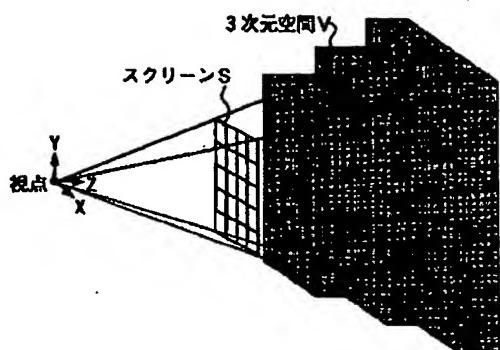
【図7】

レイトランキング手法説明図



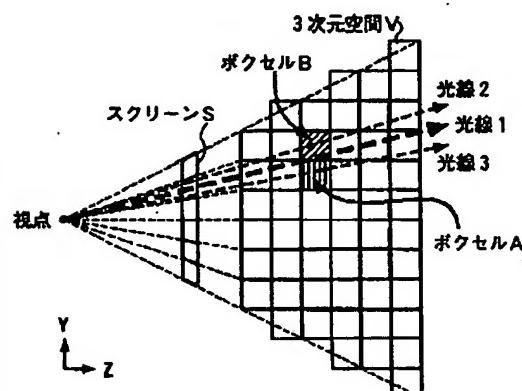
【図8】

## 従来のポクセル分割手法



【図9】

## Y-Z平面から見た従来のポクセル分割手法



フロントページの続き

(72)発明者 北澤 仁志  
東京都新宿区西新宿三丁目19番2号 日本  
電信電話株式会社内

F ターム(参考) 5B050 BA04 BA09 BA10 EA27 EA28  
FA06  
5B080 BA03 BA04 BA05 CA08 GA06

